

PL/I-80 V1.3 COMPILATION OF: GENTAB

D: Disk Print
L: List Source Program

NO ERROR(S) IN PASS 1

NO ERROR(S) IN PASS 2

PL/I-80 V1.3 COMPILATION OF: GENTAB

```

1 a 0000 gentab: proc options (main);
2 a 0006
3 a 0006      /* generate tables for 8086 disassembler 12/23/80 */
4 a 0006      /* modified 5/14/81 */
5 a 0006
6 c 0006      declare
7 c 0006          opcnt (216) fixed (7) static initial (0,0,0,0,0),
8 c 0006          sum fixed (7),
9 c 0006          len fixed (7),
10 c 0006         line char (100) varying,
11 c 0006         infile file,
12 c 0006         outfile file,
13 c 0006         litfile file,
14 c 0006         opcode char (10) varying,
15 c 0006         i fixed (7),
16 c 0006         j fixed (15),
17 c 0006         n fixed (7),
18 c 0006         count fixed (15),
19 c 0006         chars (200) char (6) varying;
20 c 0006
21 c 0006         open file (infile) input stream title ('OP86.DAT');
22 c 0022         open file (outfile) print title ('OPTAB.DAT');
23 c 003E         open file (litfile) print title ('OPTAB.LIT');
24 c 005A
25 c 005A         on endpage (outfile) begin; end;
26 c 0065         on endpage (litfile) begin; end;
27 d 0071
28 c 0071         count = 0;
29 c 0078
30 c 0078         /* read op86.dat file into chars array */
31 c 0073
32 c 0078         get file (infile) list (opcode);
33 c 0095         do while (opcode ^= '$');
34 c 00A3             count = count + 1;
35 c 00AA             chars (count) = opcode;
36 c 00C1             get file (infile) list (opcode);
37 c 00E1         end;
38 c 00E1
39 c 00E1         /* create ascii opcode tables, 1 for each character length */
40 c 00E1
41 c 00E1         do i = 2 to 6;
42 c 00F6             line = 'declare ops' !! deblank (i) !! ' (*) byte initial (';
43 c 0118             n = 0;
44 c 011C             do j = 1 to count;
45 c 0135                 if length (chars (j)) = i then
46 c 0158                     do;

```

CR/m-86 v.1.1 (Vol. III)
 COPYRIGHT © 1981, 1982
 DIGITAL RESEARCH
 P. O. BOX 579
 PACIFIC GROVE, CA 93950
 SER. # C81-162

```

47 c 0158          if n > 0 then line = line || ', '
48 c 016F          if divide (n, 5, 7) * 5 = n then
49 c 019D              do;
50 c 019D                  put file (outfile) skip list (line);
51 c 018A                  line = '^I';
52 c 01C7              end;
53 c 01C7          n = n + 1;
54 c 01CB          line = line || '""' || chars (j) || '""';
55 c 0203          opcnt (i) = opcnt (i) + 1;
56 c 021B          end;
57 c 021B          end;
58 c 021B          line = line || ')';
59 c 022A          put file (outfile) skip list (line);
60 c 0247          put file (outfile) skip;
61 c 0262          end;
62 c 0262
63 c 0262          /* create array containing # of opcodes of each length */
64 c 0262
65 c 0262          line = 'declare nops (5) byte public initial (';
66 c 026F          do i = 2 to 6;
67 c 0284              line = line || deblank (opcnt (i));
68 c 02A2              if i < 6 then line = line || ', '
69 c 02C8          end;
70 c 02C8          put file (outfile) skip list (line || ')';
71 c 02ED          put file (outfile) skip;
72 c 0301
73 c 0301          /* create array containing starting index for each opcode length */
74 c 0301
75 c 0301          line = 'declare opn$in (#) byte public initial (';
76 c 030E          sum = 0;
77 c 0312          do i = 2 to 6;
78 c 0327              line = line || deblank (sum) || ', '
79 c 0347              sum = sum + opcnt (i);
80 c 0363          end;
81 c 0363          put file (outfile) skip list (line || '255)');
82 c 0383
83 c 0388          /* create literals for starting indexes for each opcode length */
84 c 0383
85 c 0388          sum = 0;
86 c 038C          put file (litfile) skip list ('declare');
87 c 03AB          do i = 2 to 6;
88 c 03C0              put skip list (deblank (opcnt (i)), deblank (i) || '-character opcodes');
89 c 03FF              line = '^I' || 'op' || deblank (i) ||
90 c 0443                  'tin literally '""' || deblank (sum) || '""';
91 c 0443              if i = 6 then line = line || ')';
92 c 045D              else line = line || ', '
93 c 046C              put file (litfile) skip list (line);
94 c 0489              sum = sum + opcnt (i);
95 c 049E              opcnt (i) = 0;
96 c 04B4          end;
97 c 04B4
98 c 04B4          /* create literals for position in opcode tables of each opcode */
99 c 04B4
100 c 04B4          put file (litfile) skip;
101 c 04C3          put file (litfile) skip list ('declare');
102 c 04E7          do j = 1 to count;
103 c 0500              len = length (chars (j));
104 c 0516              if index (chars (j), ' ') > 0 then
105 c 0534                  chars (j) = substr (chars (j), 1, len-1);
106 c 056B              line = '^I' || chars (j) || '$in literally '

```

COPYRIGHT © 1981, 1982

DIGITAL RESEARCH

P. O. BOX 579

PACIFIC GROVE, CA 93950

SER. # _____

```

107 c 05E1      ;; 'op' ;; deblank (len) ;; '$in + '
108 c 05E1      ;; deblank (opcnt (len)) ;; ''';
109 c 05E1      if j = count then line = line ;; ' ';
110 c 0603      else line = line ;; ',';
111 c 0612      put file (litfile) skip list (line);
112 c 062F      opcnt (len) = opcnt (len) + 1;
113 c 0647      end;
114 c 0647
115 c 0647      deblank: proc (i) returns (char (10) varying);
116 e 0647      declare i fixed (7);
117 e 0651      declare temp char (10) varying;
118 e 0651      temp = char (i);
119 e 0666      return (substr (temp, verify (temp, ' ')));
120 c 0687      end deblank;
121 c 0687
122 a 0687      end gentab;

```

CODE SIZE = 068A
 DATA AREA = 06DE
 FREE SYNS = 1262
 END COMPILATION

COPYRIGHT © 1981, 1982 DDM - RESEARCH P.O. BOX 579 PACIFIC GROVE, CA 93950 SER. # _____
